

4 ENGENHARIA DE SOFTWARE PARA WEB

A disciplina de Engenharia de *Software* para *Web* foi ministrada no 3º semestre e prepara os alunos para a criação de aplicações eficientes, seguras e escaláveis para a *internet*. A importância dessa matéria está ligada à crescente demandas por aplicações *web* no mercado já que, empresas de diferentes setores dependem cada vez mais de soluções digitais para otimizar e melhorar a experiência do usuário. Por isso, o profissional precisa de uma base sólida em programação e conceitos fundamentais para o desenvolvimento de sistemas mais modernos.

Compreender o funcionamento de aplicações *web* faz com que o aluno construa sistemas mais robustos, com fácil manutenção e capazes de evoluir com as necessidades do mercado. A utilização de metodologias de desenvolvimento de *software* é essencial a fim de garantir a eficiência, qualidade e organização de projetos, exigindo uma abordagem estruturada e colaborativa.

A matéria abordou tópicos como uma metologia orientada do início ao fim, desde o levantamento de requisitos até a entrega da solução final. No caso prático, é importante compreender as particularidades de um projeto e da aplicação *web* para que o desenvolvimento seja bem sucedido, o mapeamento de modelos de representação torna-se uma etapa importante, principalmente quando se utiliza a *Unified Modeling Language (UML)*. A *UML* é uma linguagem gráfica que oferece uma notação visual padronizada para descrever o *design* de sistemas. É através de diagramas, como o diagrama de casos de uso, diagramas de classes, diagramas de sequência e diagramas de atividades, que é possível representar de forma clara a estrutura e os processos de uma aplicação *web*. Esses diagramas ajudam a entender o funcionamento do sistema e também são ferramentas importantes para a comunicação entre membros da equipe de desenvolvimento e partes interessadas no projeto.

Durante as aulas foi abordado também sobre a documentação gerada a partir da aplicação de metodologias de desenvolvimento, juntamente com os diagramas *UML*, oferece uma base sólida para possíveis futuras manutenções, melhorias ou integrações do sistema. Essa documentação serve como um guia para o usuário ou a equipe de desenvolvimento e também para outros profissionais que possam precisar entender ou modificar o sistema no futuro.

Além disso, ela é essencial para garantir que os requisitos foram atendidos e que o sistema foi construído conforme as expectativas do cliente ou usuário final.

Ao empregar uma metodologia estruturada e utilizar *UML* para mapear os modelos de representação de uma aplicação *web*, a equipe de desenvolvimento consegue planejar e executar um projeto com maior clareza, reduzindo os riscos de falhas no processo, otimizando o tempo de desenvolvimento e garantindo que o produto final seja eficiente, escalável e de fácil manutenção. A combinação de uma metodologia bem definida, o uso de *UML* para representação e a documentação detalhada resultam em um desenvolvimento de *software* mais organizado, seguro e alinhado com as necessidades do cliente.

Foram estudados os processos de *softwares*, Diagrama de Uso e sobre Planejamento de Recursos Empresariais. O objetivo é explorar conceitos fundamentais, metologias e as melhores práticas adotadas pelas organizações a fim de otimizar o uso de deus recursos, garantido eficácia, redução de custos e estratégias com seus objetivos.

4.1 Processos de Softwares

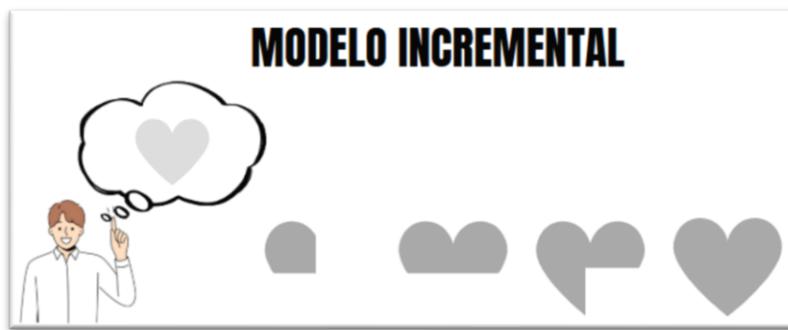
Os processos de software são fundamentais para orientar o desenvolvimento de sistemas de forma estruturada e eficiente. Eles definem as etapas, atividades e métodos que guiam desde o planejamento até a entrega e manutenção de um software, garantindo qualidade, confiabilidade e alinhamento com as necessidades do cliente. A escolha do modelo de processo adequado depende de fatores como o tipo de sistema, a complexidade do projeto e o nível de flexibilidade exigido. Entre os principais modelos utilizados estão o modelo cascata, o modelo incremental e o modelo iterativo.

4.1.1 Modelos

Um dos primeiros assuntos abordados foi a escolha de qual modelo de processo de *software* considerar de acordo com as particularidades de cada sistema que está sendo desenvolvido. Durante o desenvolvimento de *software*, a escolha do modelo de processo adequado é essencial a fim de garantir o sucesso do projeto. Assim, devemos pensar nos fatores como o tipo de sistema, a complexidade, os riscos envolvidos e o nível de flexibilidade necessário. Entre os principais modelos estudados nas aulas estão: o modelo incremental, o modelo cascata e o modelo iterativo e incremental.

O Modelo Incremental baseia-se na construção do *software* em pequenas partes funcionais (incrementos), que são desenvolvidas, testadas e entregues de forma progressiva, permitindo um *feedback* constante, e facilitando a identificação de falhas, sendo o ideal para sistemas que exigem evolução contínua ou validações frequentes, como sistemas críticos ou inovadores, e que garantam segurança e confiabilidade, tornando o modelo incremental uma escolha mais adequada por permitir testes frequentes e entregas controladas. Esse modelo também se adequa a projetos inovadores, como sistemas de realidade virtual voltados à manutenção de *software*, pois oferece flexibilidade, entregas contínuas e adaptação a mudanças. A imagem abaixo ilustra como o modelo é criado em partes e desenvolvido:

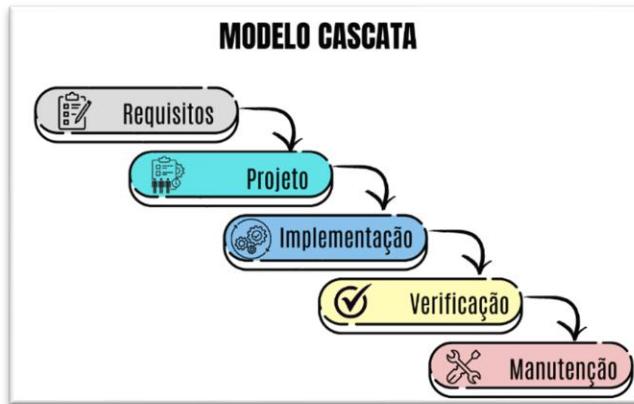
Figura 1- 4.1 - Exemplo do Modelo Incremental



Fonte: Elaborada pela autora, 2024.

O próximo é o Modelo Cascata, que segue uma abordagem sequencial, com fases bem definidas como requisitos, análise, *design*, implementação, testes e manutenção. É importante salientar que cada fase precisa ser concluída antes do início da próxima. Este modelo é mais indicado quando os requisitos estão bem compreendidos desde o início, como em sistemas que substituem soluções já existentes, pois segue uma sequência bem definida de etapas, facilitando o controle do processo e garantindo uma transição estruturada como na imagem a seguir:

Figura 2 - 4.2 - Estrutura do Modelo Cascata



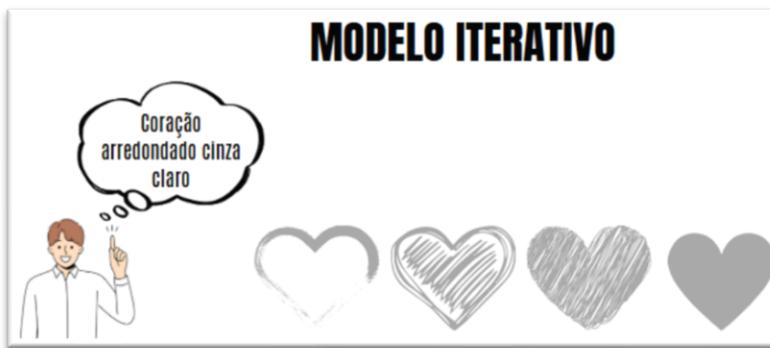
Fonte: Elaborada pela autora, 2024.

O modelo iterativo se destaca por sua flexibilidade e capacidade de adaptação às mudanças ao longo do projeto. Ao contrário do modelo em cascata, que segue uma sequência linear de etapas, o modelo iterativo propõe o desenvolvimento do sistema por meio de ciclos repetidos, conhecidos como iterações. Cada iteração resulta em uma versão funcional do software, ainda que incompleta, que pode ser testada e validada.

Essa forma de desenvolvimento permite que os requisitos do sistema sejam refinados continuamente, com base no *feedback* obtido a cada nova entrega. Durante uma iteração, são realizadas atividades de planejamento, levantamento de requisitos, projeto, codificação e testes, o que garante uma evolução constante do produto. Esse ciclo se repete até que o sistema esteja todo estruturado e com funcionalidade suficiente para ser considerado concluído.

Com o modelo iterativo, é possível detectar e corrigir falhas nas fases iniciais do projeto, reduzindo o risco de retrabalho e prejuízos no final do processo. Além disso, o envolvimento constante do cliente e das partes interessadas favorece a construção de um *software* mais alinhado com as expectativas do usuário final. A entrega gradual de partes funcionais também permite que o cliente acompanhe o progresso e valide o produto de forma contínua como visto na imagem abaixo:

Figura 3 - 4.3 - Exemplo de Modelo Iterativo



Fonte: Elaborada pela autora, 2024.

Por fim, o modelo iterativo e incremental que combina as vantagens do desenvolvimento incremental com ciclos de repetição (iterações). A cada ciclo, o sistema é analisado, desenvolvido, testado e melhorado com base no *feedback*. É ideal para sistemas interativos e sujeitos a mudanças constantes, pois permite refinar o *software* ao longo do tempo pois permite incorporar *feedback* dos usuários e adaptar o sistema ao longo do desenvolvimento.

4.1.2 Princípios e Atributos

A engenharia de *software* é uma área ampla e dinâmica, que exige um conjunto de atividades contínuas e essenciais, como suporte técnico, atualizações regulares, documentação clara, treinamentos e possibilidades de personalização que evoluam conforme as novas demandas ou mudanças.

É importante diferenciar dois tipos principais de *software*: o genérico e o personalizado. O *software* genérico é desenvolvido com o objetivo de atender a um grande número de usuários, oferecendo funcionalidades amplas e padronizadas podendo não suprir completamente as necessidades específicas de cada cliente. Já o *software* personalizado é projetado sob medida, de acordo com as exigências e características de um cliente ou organização.

Para garantir a qualidade de um sistema, diversos atributos de *software* devem ser considerados. Os quatro principais são: a confiabilidade, que assegura o funcionamento correto do sistema; a segurança, essencial para proteger dados e operações; o desempenho, que diz respeito à eficiência do sistema durante o uso; e a escalabilidade, que garante que o *software* possa

crescer conforme as demandas e a necessidade. Além desses, outros atributos igualmente relevantes são a usabilidade, que é a facilidade de uso, a flexibilidade e capacidade de adaptação, a manutenibilidade na facilidade para manutenção e correções e por fim, a interoperabilidade que é a capacidade de integração com outros sistemas.

A escolha das técnicas e abordagens adequadas variam conforme o tipo de aplicação. Sistemas em tempo real, como vídeo *games* ou *softwares* industriais, precisam de respostas imediatas e confiáveis, exigindo arquiteturas e linguagens que priorizem o desempenho. Enquanto, aplicações de inteligência artificial, como assistentes virtuais, dependem de algoritmos avançados e aprendizado contínuo, demandando maior capacidade de processamento e análise de dados.

Alguns princípios fundamentais são universais na engenharia de *software*, como a definição de um processo estruturado, a busca por dependabilidade, o gerenciamento de requisitos e o reuso de componentes. Esses princípios servem como base a fim de garantir que o desenvolvimento seja eficiente, seguro e alinhado aos objetivos do cliente e dos usuários finais. A necessidade de acesso remoto, interfaces intuitivas, segurança robusta e distribuição de serviços em diferentes plataformas tornou necessário arquiteturas mais flexíveis e à popularização de metodologias ágeis. A interoperabilidade e a escalabilidade tornaram-se aspecto fundamentais especialmente em sistemas *web* e *mobile*, onde a capacidade de integrar diferentes tecnologias é essencial.

É também papel do engenheiro de *software* garantir que haja transparência, responsabilidade social, limites bem definidos e garantias que protejam a privacidade e os direitos individuais dos cidadãos, respeitando os valores éticos e legais que regem a sociedade.

4.1.3 Requisitos

Os requisitos representam um dos pilares fundamentais e descrevem o que o sistema deve fazer, são os requisitos funcionais, e como ele deve se comportar. Os requisitos não-funcionais, além de condições ou restrições impostas ao desenvolvimento. Os requisitos funcionais definem as funções específicas que o sistema deve realizar e os requisitos não funcionais

estabelecem critérios de qualidade do sistema, como desempenho, segurança, usabilidade, confiabilidade, entre outros.

A fase do levantamento de requisitos é uma das mais críticas do processo de desenvolvimento de *software* pois, quando os requisitos são mal definidos ou incompletos, aumenta-se os riscos de falhas no projeto, atrasos, retrabalho, despesas extras e insatisfação do cliente. Por isso, é essencial que essa etapa envolva uma boa comunicação entre os desenvolvedores e os clientes.

Existem algumas atividades relacionadas ao requisitos que ajudam no processo, como a elicitação, ou seja, o processo de coletar as necessidades do cliente, através de entrevistas, questionários, observações e reuniões; a elaboração e análise por meio da viabilidade, consistência e completude dos requisitos; a negociação das partes, a especificação de documentação formal dos requisitos em linguagens comprehensíveis para as partes envolvidas; a validação e verificação de que os requisitos realmente correspondem às necessidades do usuário e o gerenciamento de requisitos através do controle das mudanças nos requisitos ao longo do ciclo de vida do *software*. Na imagem abaixo podemos observar os requisitos:

Figura 4 - 4.4 - Requisitos de um Projeto



Fonte: Elaborada pela autora, 2024.

4.1.4 Programa Astah

O *Astah* é uma ferramenta de modelagem visual e seu objetivo é auxiliar na criação, documentação e compreensão de sistemas por meio de diagramas gráficos. A modelagem de *software* é uma etapa essencial no desenvolvimento e permite representar graficamente os elementos do sistema, facilitando tanto o planejamento quanto a comunicação entre os envolvidos no projeto.

Entre os principais recursos oferecidos pelo *Astah* estão os diagramas de casos de uso, que ajudam a identificar e representar as funcionalidades do

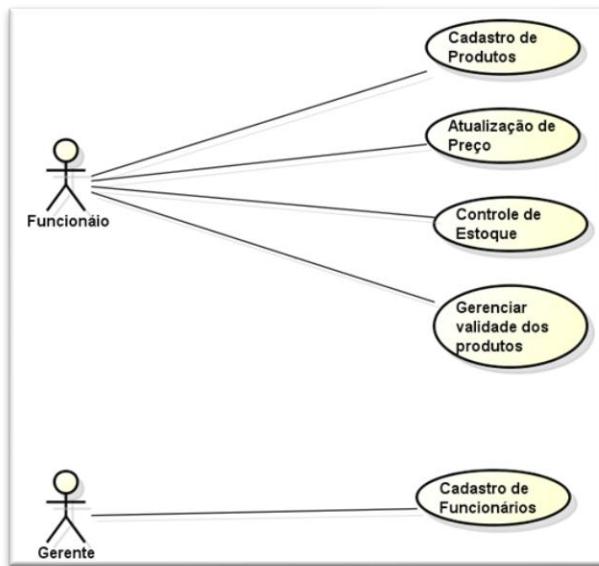
sistema do ponto de vista do usuário; diagramas de classes, que modelam a estrutura estática do sistema; e diagramas de atividades e sequência, que ilustram o fluxo de processos e a interação entre os objetos ao longo do tempo. Permite criar diagramas de estado, componentes e implantação, cobrindo necessidades no *design* de sistemas complexos.

4.1.4 Diagrama de Casos de Uso - Mercearia Quitanda do Bairro

O problema do exercício era o fato de a mercearia estar em expansão com aumento no número de produtos e clientes, o que gerou a necessidade de contratar mais funcionários e melhorar o controle de estoque, validade dos produtos e preços. Atualmente, esses processos são manuais, dificultando a gestão. Um sistema automatizado e integrado ao caixa é necessário para facilitar essas tarefas.

O diagrama abaixo mostra as funcionalidades (casos de uso) que o sistema oferecerá e como os atores (usuários do sistema) interagem com essas funcionalidades:

Figura 5 - 4.5 - Atores e suas Funcionalidades



Fonte: Elaborada pela autora, 2024.

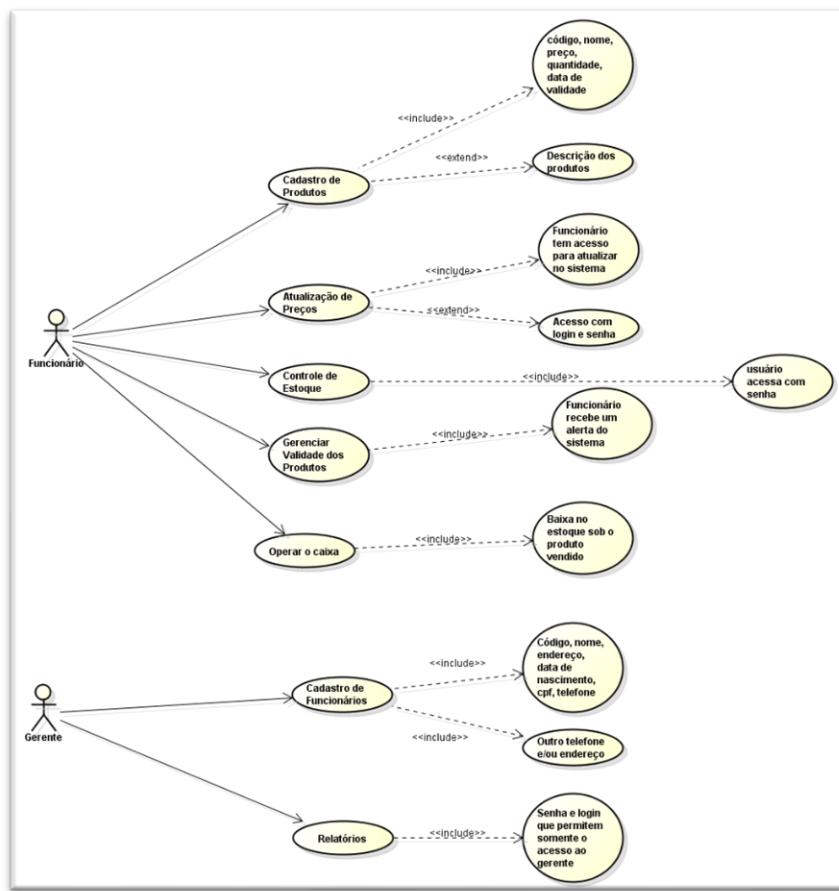
Nesse sistema existem dois perfis principais de usuários: o Funcionário e o Gerente. O Funcionário representa os usuários que utilizam o sistema no dia a dia da operação da mercearia. Isso inclui operadores de caixa, repositores de

produtos e responsáveis pelo controle de estoque. Esses usuários têm permissões limitadas que são restritas a tarefas operacionais, como o cadastro e atualização de produtos, controle de estoque, gerenciamento de validade dos produtos e operação do caixa.

Já o Gerente é o usuário com maior nível de acesso dentro do sistema como administrador. Ele tem acesso a informações sensíveis e funções administrativas, como o cadastro de novos funcionários e a geração de relatórios gerenciais. Essas funcionalidades são protegidas por *login* e senha, garantindo que apenas o gerente possa acessá-las, assegurando a segurança e a integridade dos dados gerenciados.

Na imagem a seguir temos as funções e os casos de uso representados pelos funcionários e pelo gerente:

Figura 6 - 4.6 - Casos de Uso



Fonte: Elaborada pela autora, 2024.

Entre as principais funções operacionais acessadas pelos funcionários, estão o Cadastro de Produtos, Atualização de Preços, Controle de Estoque, Gerenciamento da Validade dos Produtos e a Operação do Caixa. Todas essas

atividades são restritas a usuários autenticados, exigindo *login* e senha para garantir a segurança das informações.

A funcionalidade de Cadastro de Produtos permite que os funcionários registrem novos itens no sistema, inserindo informações fundamentais como código, nome, preço, quantidade e data de validade. Esse processo é essencial para manter o banco de dados de produtos atualizado e confiável, garantindo que todos os itens comercializados estejam devidamente registrados. Essa ação está condicionada ao *login* prévio do funcionário, respeitando as regras de segurança do sistema.

A Atualização de Preços é uma funcionalidade que possibilita a alteração dos valores dos produtos cadastrados, o sistema necessita que o funcionário esteja autenticado e autorizado para realizar essa modificação, assim o sistema previne alterações indevidas, mantendo o controle sobre os valores praticados pela mercearia.

O Controle de Estoque permite que o funcionário visualize a quantidade atual de produtos disponíveis e realize ajustes de entrada ou saída. Essa funcionalidade é fundamental para evitar problemas de estoque ou o excesso de mercadorias armazenadas. Assim como as outras funções, ela também exige autenticação do usuário por meio de *login* e senha.

Com o objetivo de garantir a qualidade dos produtos oferecidos aos clientes, o sistema conta com o recurso de “Gerenciar Validade dos Produtos”. Essa função monitora as datas de validade dos itens cadastrados e emite alertas automáticos para o funcionário quando um produto estiver prestes a vencer ou já estiver vencido. Isso auxilia na retirada preventiva de mercadorias impróprias para o consumo, preservando a reputação da mercearia e a saúde dos consumidores.

Por fim, a funcionalidade de operar o Caixa permite que o funcionário registre vendas e finalize compras dos clientes. Além de registrar os pagamentos, essa função está diretamente integrada ao controle de estoque: a cada venda realizada, o sistema faz automaticamente a baixa do produto vendido no estoque. Essa integração entre as funções é crucial para manter a consistência das informações e garantir uma operação fluida e precisa.

Além das funcionalidades voltadas aos funcionários operacionais, o sistema também contempla recursos exclusivos para o gerente, o qual é a figura

responsável pela administração geral do estabelecimento. Com acesso superior, o gerente pode executar tarefas mais sensíveis, como o “Cadastro de Funcionários” e a geração de “Relatórios Gerenciais”. O acesso a essas funções é protegido por autenticação específica, garantindo que apenas usuários com perfil de gerência possam executá-las.

A funcionalidade do Cadastro de Funcionários permite que o gerente registre novos colaboradores no sistema. Durante esse processo, são inseridos dados importantes, como código, nome, endereço, data de nascimento, CPF e telefone. Além disso, o gerente tem a possibilidade de complementar ou atualizar essas informações posteriormente, conforme necessário. Esse controle é essencial para manter o cadastro de pessoal organizado e atualizado, facilitando a gestão da equipe.

Outra função essencial sob responsabilidade do gerente é a Geração de Relatórios. Esses documentos oferecem uma visão consolidada sobre o funcionamento da mercearia, com dados sobre vendas, movimentação de estoque, validade de produtos, entre outros. Os relatórios são fundamentais para a tomada de decisões estratégicas, como reposição de mercadorias, promoções e ajustes no quadro de funcionários. Para garantir a confidencialidade das informações, o acesso a essa função também está protegido por *login* e senha exclusivos ao gerente.

Os requisitos funcionais incluem o cadastro de produtos, controle automático do estoque, atualização de preços, alertas para produtos vencidos, gestão de funcionários com níveis de acesso e integração com o caixa. Esses recursos garantem a automação e eficiência das tarefas diárias.

Já os requisitos não-funcionais asseguram que o sistema seja confiável, seguro e fácil de usar. Entre eles estão o bom desempenho mesmo em horários de pico, segurança com autenticação e proteção de dados, usabilidade intuitiva com treinamentos, disponibilidade total (24/7) e escalabilidade para acompanhar o crescimento da mercearia.

4.2 Trabalho: Planejamento de Recursos Empresariais

4.2.1 *ERP*

ERP ou Planejamento de Recursos Empresariais, é um sistema de gestão empresarial que integra e automatiza processos em uma organização que abrange áreas como contabilidade, finanças, vendas, compras e produção. Um *ERP* melhora a eficiência, reduz custos e fornece uma visão abrangente das operações em tempo real. Ele é altamente personalizável e muito utilizado em empresas de todos os tamanhos e setores. O processo de unificar os dados e operações em único sistema, permite que as empresas consigam ver em tempo real suas atividades tornando mais fácil a tomada de decisão.

Antes desses sistemas, cada departamento costumava operar seus próprios dados e resultados, o que frequentemente gerava inconsistências, dificuldade na comunicação interna, atrasos e retrabalhos. Com o sistema *ERP*, as informações são centralizadas e podem ser acessadas a qualquer momento, garantindo transparência e eficiência nas operações. Essa integração, melhora a produtividade e reduz custos operacionais.

A capacidade de análise de dados e geração de relatórios bem detalhados e aprofundados, ajuda a identificar erros, necessidades, tendências e prevê demandas a fim de otimizar o uso dos recursos da empresa. Um dos fatores essenciais dos sistemas *ERP* é escalabilidade e personalização, ou seja, à medida que for necessário aumentar a capacidade, é possível fazer essa alteração. Empresas de diferentes tamanhos com poucos ou muito setores podem adaptar o *software* para as suas necessidades específicas, escolhendo módulos e funcionalidades que melhor irão atender suas operações. Empresas pequenas podem começar com versões mais simples e ao longo da necessidade, adicionar novos recursos conforme crescem.

Atualmente, os sistemas *ERP* estão em constante evolução e adota tecnologias como inteligência artificial, análise de *big data* e computação em nuvem para tornar ainda mais acessível e eficiente para que as empresas acompanhem as mudanças do mercado e se tornem mais competitivas.

4.2.2 Arquitetura *ERP*

A arquitetura de um sistema *ERP* refere-se à estrutura e organização do sistema de *software* que suporta as operações empresariais. Ela define como os componentes do sistema estão interconectados e como as informações fluem

entre esses componentes para facilitar os processos de negócios da empresa. Nas camadas da arquitetura é possível visualizar a estrutura do sistema e como ele atende as demandas da empresa de forma organizada, escalável e segura. Na figura abaixo podemos observar quais são as camadas da arquitetura que veremos a seguir:

Figura 7 - 4.7 - Requisitos de um projeto



Fonte: Elaborada pela autora, 2024.

Na Camada de Apresentação, na *UI*, os usuários interagem com o sistema, por meio de interfaces gráficas de usuário (*GUI*) ou aplicativos móveis. Ela inclui telas, formulários e relatórios que permitem que os usuários realizem tarefas e acessem informações no sistema. Atualmente, muitas empresas preferem plataformas móveis além de *sites* com interface *web* a fim de garantir que os usuários tenham acesso ao sistema em qualquer lugar que estiver. A camada exibe informações de maneira organizada e permite que os usuários insiram dados e executem comandos no sistema.

A seguir vem a Camada de Lógica de Negócios, a Aplicação, que contém a lógica de negócios do sistema, incluindo os processos, regras e algoritmos que governam o comportamento do sistema. Aqui estão os módulos funcionais do *ERP*, como contabilidade, finanças, recursos humanos, vendas, compras, produção, entre outros. Esta camada é o “coração” do sistema *ERP* e é responsável por processar as informações de acordo com as regras de negócios específicas de cada parte da organização. Essa camada é dinâmica e adaptável de acordo com a necessidade da empresa já que a flexibilidade e escalabilidade dessa camada são cruciais para que o sistema suporte a evolução como mudanças de regras ou crescimento da empresa.

A Camada de Aplicação é dividida em módulos específicos, que são componentes funcionais separados, como gestão financeira, gestão de recursos

humanos, gestão de produção, gestão de vendas, entre outros. Cada um desses módulos interage com os outros para garantir que todas as áreas da empresa estejam em sincronia.

A Camada de Dados, são os bancos de dados que armazenam todas as informações do sistema, como dados mestre, transacionais e de configuração e utiliza um banco de dados centralizado para garantir consistência e integridade dos dados em toda a organização. As informações são mantidas em um formato estruturado, isso permite que os departamentos da empresa acessem os dados em tempo real. O banco de dados deve ser robusto para suportar o volume de dados e garantir a segurança através de *backups* e criptografia.

A próxima é a Camada de Integração, *Middleware*, que facilita a conexão entre diversos sistemas internos e externos, permitindo a troca eficiente de dados. Em muitas empresas, o *ERP* não opera sozinho, e está integrado com outros sistemas, como o *CRM*, *WMS*, Sistemas de Comércio Eletrônico e outros sistemas especializados. Esta camada é responsável para que os dados fluam continuamente entre as plataformas sem interrupções ou inconsistências. Ela facilita a automação de processos aumentando a eficácia. Atualmente, a integração com sistemas em nuvem as outras plataformas como a *API* (*Application Programming Interface*) é cada vez mais importante. O uso de *web services* e mecanismos de integração torna possível a troca de informações em tempo real, oferecendo uma visão ampla e atualizada dos processos da empresa.

Por último, a Camada de Infraestrutura, servidores e rede, são servidores físicos ou virtuais e infraestrutura de rede necessária para executar o sistema, rede de comunicação e os recursos computacionais que suportam o processamento de dados em larga escala. Envolve aspectos de segurança de informação como *firewalls*, autenticação de usuários, e outros mecanismos de proteção contra ameaças externas. Em sistemas *ERP* baseados na nuvem, a infraestrutura envolve a gestão de servidores em provedores de *cloud computing* para garantir disponibilidade e desempenho.

4.2.3 Vantagens *ERP*

A implementação de um sistema de Planejamento de Recursos Empresariais oferece benefícios estratégicos e operacionais para as empresas em decorrência da crescente complexidade das operações e da necessidade de otimizar processos. As principais vantagens são a automatização de tarefas, reduzir o tempo gasto com atividades repetitivas, redução de custo e uma gestão mais eficiente dos recursos. A integração entre os processos e a centralização de dados tornam a gestão segura. A seguir são exploradas algumas das vantagens mais significativas que um sistema *ERP* pode oferecer a uma organização:

Figura 8 - 4.8 - Vantagens do Planejamento de Recursos Empresariais



Fonte: Elaborada pela autora, 2024.

A primeira vantagem é a “Automação”, no qual um sistema de gestão integrado reduz o tempo necessário aos servidores para desempenhar tarefas burocráticas e repetitivas. O *ERP* automatiza as atividades e facilita a padronização de processos, com a adoção de estruturas simplificadas. A automação melhora a eficiência operacional e permite com que a empresa seja mais produtiva. A simplificação e padronização dos processos garantem que as práticas organizacionais sejam seguidas de forma consistente.

A próxima é a “Redução de Custos” já que o *software* tem o papel de integrar informações de diferentes departamentos da instituição, assim, ele pode fazer um diagnóstico mais aprofundado sobre as medidas necessárias para diminuir custos sem afetar a produtividade e ajuda a identificar os níveis necessários de

estoque. O *software* pode também identificar os itens mais rentáveis ou mais solicitados, permitindo um controle mais preciso sobre as compras e evitando assim desperdícios.

A “Transparência e Segurança” geradas pela *ERP* centraliza os dados corporativos, oferecendo maior transparência aos processos por meio de *dashboards* que exibem as atividades de cada área, simplificando o monitoramento das demandas de cada setor e aumentando a segurança dos dados, especialmente quando armazenados na nuvem. A segurança dos dados também é um ponto importante especialmente em um ambiente corporativo que informações confidenciais, como dados financeiros e pessoais de clientes, assim, um *ERP* bem estruturado garante a proteção dessas informações por meio de criptografia, controles de acesso, *backups* regulares e, muitas vezes, a utilização de servidores na nuvem.

A “Diminuição de Erros” no sistema é outra vantagem proporcionada pelo *ERP*. Como citado anteriormente, quando as informações são inseridas de maneira manual transferidas de um sistema para o outro, há o risco de erros tipo erros de cálculo, duplicidade de dados, inconsistências e outros. Com o processo automatizado é possível a redução de falhas no gerenciamento e problemas de informações. Além disso, os sistemas *ERP* possuem ferramentas de validação de dados, que garantem que apenas informações corretas e completas sejam registradas, aumentando a confiabilidade das informações e o desempenho geral da empresa.

“Gestão de Pessoas” é a próxima vantagem que, por meio de módulos específicos o sistema pode otimizar processos como avaliação de desempenho, recrutamento, folha de pagamento, benefícios e treinamentos.

Por fim, o “Auxílio às Tomadas de Decisão” que detectam falhas nos processos e a visualização dos principais fatores que geram despesas desnecessárias. O gestor acompanha os *KPIs*, analisa e identifica as principais causas dos problemas, e adota medidas para solucioná-los com mais rapidez e eficiência. Com dados centralizados e em tempo real, os gestores podem tomar decisões mais rápidas e embasadas. O *ERP* fornece relatórios detalhados e gráficos que ajudam a identificar pontos críticos nos processos.

4.2.4 Empresas que usam *ERP*

Grandes corporações têm adotado diferentes soluções *ERP*, cada uma escolhendo o sistema que melhor se adapta às suas necessidades específicas. Entre os mais usados estão o *SAP ERP*, o *Oracle ERP Cloud*, o *Microsoft Dynamics 365*, o *Infor ERP*, o *NetSuite ERP* e o *Epicor ERP*.

O *SAP ERP*, desenvolvido pela *SAP SE*, é um dos sistemas mais usados no mundo. Empresas como *Procter & Gamble* e *Coca-Cola* dependem desse sistema para integrar suas operações globais, abrangendo desde a gestão de suprimentos até as finanças. A flexibilidade e robustez do *SAP* permitem que empresas grandes gerenciem suas diversas operações com eficiência, além de oferecerem um alto nível de personalização para atender às necessidades específicas de cada setor.

Outro grande nome no mercado de *ERP* é o *Oracle ERP Cloud* e é muito utilizado por empresas que precisam de soluções baseadas em nuvem. O *Walmart*, por exemplo, utiliza o sistema para integrar suas operações em todo o mundo, gerenciando inventários, cadeia de suprimentos, vendas e logística simultaneamente em tempo real. O Banco do Brasil também usa para gerir suas operações financeiras e administrativas em grande escala, otimizando suas funções internas e aumentando sua produtividade.

O *Microsoft Dynamics 365* é uma solução de *ERP* e *CRM* pertencente a *Microsoft*, e se destaca por sua capacidade de integrar diversos processos de negócios em uma única plataforma. Grandes empresas como a *Microsoft* e a *Toyota* fazem uso dele para gerenciar finanças, recursos humanos, vendas e serviços em uma estrutura ágil e integrada. Este sistema é vantajoso para empresas que já estão imersas no ecossistema *Microsoft*, garantindo uma fácil integração com outras ferramentas da empresa.

O *NetSuite ERP*, também pertence a *Oracle* e é uma solução baseada em nuvem, que permite a integração e a gestão de processos de negócios, sendo ideal para empresas em crescimento. A *Amazon* utiliza o *NetSuite* para gerenciar suas operações de varejo, controle de estoque, pedidos e cadeia de suprimentos. A flexibilidade da plataforma oferece uma solução escalável que pode crescer junto com a empresa.

O *Epicor ERP* é mais voltado para empresas de manufatura, distribuição, varejo e serviços. *Ford* e *Boeing*, por exemplo, utilizam esse sistema para otimizar suas operações de produção e logística em uma escala global.

A *Apple* utiliza sistemas *ERP* para gerenciar sua cadeia de suprimentos global, além das operações de fabricação, distribuição e vendas. A *ExxonMobil*, do setor de energia, também adota sistemas *ERP* para coordenar suas operações de exploração, produção, refino e distribuição de petróleo e gás. O *McDonald's*, utiliza o *ERP* para gerenciar suas operações de restaurantes, incluindo o controle de estoque, compras, vendas e recursos humanos em nível global.

4.2.5 Módulos *ERP*

O uso do *ERP* permite que essas organizações mantenham uma visão unificada e em tempo real de todos os seus processos, resultando em uma gestão mais eficiente e estratégica. A seguir veremos quais são seus módulos:

Figura 9 - Figura 4.9 - Módulos do Planejamento de Recursos Empresariais



Fonte: Elaborada pela autora, 2024.

O sistema ERP é composto por diferentes módulos que se integram para proporcionar uma gestão completa e estratégica dentro das organizações. O “Controle Financeiro” é responsável por registrar transações, acompanhar contas a pagar e receber, elaborar orçamentos, controlar custos, gerir ativos e gerar relatórios que apoiam a saúde financeira da empresa. O “Controle de

Estoque” permite monitorar entradas e saídas de mercadorias, gerenciar a localização dos itens, automatizar reabastecimentos, oferecer rastreabilidade de produtos e realizar análises que evitam desperdícios, garantindo maior eficiência no atendimento da demanda. A “Emissão de Notas Fiscais e Boletos” assegura a conformidade fiscal, automatiza o cálculo de impostos, possibilita a criação de documentos personalizados e facilita o controle de contas a receber por meio da integração com instituições financeiras.

O “PDV Online” (Ponto de Venda) integra vendas e estoque em tempo real, processa pedidos com agilidade, registra o histórico de clientes e gera relatórios que contribuem para estratégias de *marketing* e programas de fidelização. Já o “WebDesk – Portal do Cliente” oferece acesso 24 horas a informações como pedidos e faturas, possibilita autosserviço, centraliza atendimentos e mantém integração direta, aumentando a autonomia e a confiança do consumidor. Os “Relatórios de Processos” apresentam indicadores de produtividade, qualidade e *compliance*, permitindo a análise de desempenho, a identificação de falhas e a antecipação de tendências.

Outro recurso importante é a “Conciliação Bancária”, que automatiza a importação de extratos, confere transações, identifica discrepâncias e gera relatórios financeiros, garantindo maior precisão e segurança. Por fim, o “Cadastro de Clientes e Funcionários” organiza informações pessoais, históricos de compras, crédito e segmentação de clientes, além de armazenar dados trabalhistas, remuneração, desempenho e permissões de acesso dos colaboradores, contribuindo para uma gestão mais estruturada e eficiente.

4.2.6 Considerações Finais

Implementar um sistema *ERP* é uma forma estratégica em busca de otimizar seus processos, aumentar a eficiência e otimizar as decisões. Com a integração das áreas permite uma visão unificada e em tempo real de todas as operações contribuindo para a redução de custos, melhora a segurança e integridade dos dados, facilita as regulamentações fiscais, de estoque e contábeis.